

Report to

WA State Office of Financial Management

Grants, Contracts and Loans Feasibility Study

Indicative Function Point Count

Version 1.0

Sierra Systems
1177 West Hastings Street, Suite 2500
Vancouver, BC V6E 2K3 Canada
www.SierraSystems.com

Contact: Barry Truter
Phone: 604.688.1371
Fax: 604.688.6482
Email: Barry.Truter@SierraSystems.com



Date: February 9, 2006

TABLE OF CONTENTS

| | |
|---|----|
| 1. Management Summary | 1 |
| 2. Introduction..... | 2 |
| 2.1. Document Purpose | 2 |
| 2.2. References..... | 2 |
| 2.3. Approach..... | 2 |
| 3. Function Point Analysis..... | 4 |
| 3.1. What is Function Point Analysis?..... | 4 |
| 3.1.1. Determine the Type of Count | 5 |
| 3.1.2. Identify the Counting Scope and Application Boundary | 5 |
| 3.1.3. Determine the Unadjusted Function Point Count..... | 5 |
| 3.1.4. Determine the Value Adjustment Factor | 6 |
| 3.1.5. Calculate the Adjusted Function Point Count | 6 |
| 3.2. Types of Function Point Counts and their Accuracy | 6 |
| 3.2.1. Indicative Function Point Count..... | 6 |
| 3.2.2. Estimated Function Point Count..... | 7 |
| 3.2.3. Detailed Function Point Count | 7 |
| 4. Indicative Function Point Count | 8 |
| 4.1. Indicative Function Point Count Type..... | 8 |
| 4.2. Application Boundary and Scope | 8 |
| 4.3. Assumptions and Notes..... | 8 |
| 4.4. Function Point Summary | 9 |
| 5. Conclusion | 11 |

Appendices

APPENDIX A. INDICATIVE FUNCTION POINT COUNT SPREADSHEET

Confidentiality/Validity

This document has been prepared by Sierra Systems for the sole purpose and exclusive use of the Washington State Office of Financial Management. Due to the confidential nature of the material in this document, its contents should not be discussed with, or disclosed to, third parties without the prior written consent of the Washington State Office of Financial Management.

1. MANAGEMENT SUMMARY

Function Point Analysis (FPA) is a method for measuring the functionality provided to a user by software. The functionality measured is only that specifically requested by the user, and it is quantified independently of the technology used to develop it. The many benefits of FPA include that it provides a standard measure for comparing software size and is helpful in estimating the amount of time it will take to develop applications.

FPA can be performed at different points throughout the software design and development process. The three primary types of function point counts are indicative, estimated and detailed counts. An indicative count is performed early on in the project and is based on either a conceptual data model or third normal form data model. An estimated count is performed when more information is available on the transactions performed by the system and its general system characteristics (e.g., operational ease requirements, prevalence of distributed data processing, etc.). When the data model is complete and all the transactions are defined to a level of detail that includes all the data elements involved and files referenced, a detailed count can be performed. The level of accuracy of the count increases from the least accurate indicative count (error margin of up to 50%) to the most accurate detailed count.

The following report outlines the results of the indicative function point count for the Washington State Office of Financial Management (OFM) Grants, Contracts and Loans (GCL) System. This count is based on the information in the GCL Definition of Requirements report dated February 7, 2006.

The functionality counted includes the functionality listed in the GCL Definition of Requirements Appendix C (34 Use Cases).

Several important assumptions were made during the count. These assumptions are detailed in Section 4.3 of this document. If these assumptions are incorrect, the function point count will be affected.

The base indicative function point count is 1,185. The level of accuracy for an indicative count is plus/minus 50% giving a FPC range of 593 to 1,778. Given the anticipated high level of business, data and workflow complexity inherent in this type of system, and the estimating risks and assumptions presented in Section 4.3 of this report, it is recommended that the lower range figure be discounted, and the indicative FPC range formalized at 1,185 to 1,778 function points.

2. INTRODUCTION

The following report outlines the results of the indicative function point count for the Washington State Office of Financial Management (OFM) Grants, Contracts and Loans (GCL) System. This count is based on the information in the GCL Definition of Requirements report dated February 7, 2006.

2.1. Document Purpose

The objectives of the indicative function point count report are:

- Present the results of the indicative function point count
- Outline the scope and functionality included in the indicative function point count
- List any assumptions made during the indicative function point count
- Describe the purpose of function point analysis and how it is performed

2.2. References

The following documents were reviewed during the preparation of the Indicative Function Point Count:

- Washington State Office of Financial Management (OFM) Grants, Contracts and Loans (GCL) System Definition of Requirements report dated February 7, 2006
- IFPUG Function Point Counting Practices Manual Release 4.1.1
- NESMA Definition and Counting Guidelines for the Application of Function Point Analysis Version 2.0

2.3. Approach

To perform the indicative function point count, the following steps were performed:

- The information in the GCL System Definition of Requirements report was analyzed.
- A conceptual data model consisting of a list of conceptual business and system entities was derived from the 34 Use Cases published in Appendix C of the Definition of Requirements report.

- Clarifications on application data and functionality were obtained from the Sierra Systems GCL Feasibility Study team.
- An indicative function point count was performed.
- The indicative function point count was reviewed with the Sierra Systems GCL Feasibility Study team. Comments on the functionality counted and assumptions were incorporated into the function point count.
- The results of the indicative function point count were provided to the Sierra Systems GCL Feasibility Study team for estimating purposes.

3. FUNCTION POINT ANALYSIS

3.1. What is Function Point Analysis?

As defined by the International Function Point Users Group, Function Point Analysis (FPA) "measures software by quantifying the functionality the software provides to the user based primarily on logical design". The functionality measured is only that specifically requested and received by the user, and it is quantified independently of the technology used to develop it.

The benefits of using FPA are numerous and include:

- Help estimate the time it will take to develop and maintain software
- Unit of measure to support quality and productivity analysis
- Standard measure for comparing software
- Determine the size of a packaged application
- Determine the benefit of software to an organization by counting functions that specifically match requirements

FPA does not automatically deliver error free project estimates but it does provide important support in the project budgeting process. This is one of the many reasons Sierra Systems has adopted function point analysis as yet another way of making the software development process more reliable for our clients.

There are five steps involved in performing FPA. **(Note that only steps 1, 2 and 3a are applicable to an indicative function point count.)**

1. Determine the type of count
2. Identify Counting Scope and Application Boundary
3. Determine the Unadjusted Function Point Count
 - a) Count Data Functions (ILFs and EIFs)
 - b) Count Transactional Functions (EIs, EOs and EQs)
4. Determine the Value Adjustment Factor
5. Calculate the Adjusted Function Point Count

3.1.1. Determine the Type of Count

There are several types of function point counts. A Development Function Point Count measure the functionality delivered with the first installation of the software. An Enhancement Project Function Point Count measures functionality that is added, changed or deleted during modifications to existing software. An Application Function Point Count is a measure of the current functionality provided to users. The first application count for a software system is called the Baseline Count. The Baseline Count is updated every time there is an enhancement project.

3.1.2. Identify the Counting Scope and Application Boundary

The counting scope determines the functionality that will be included in the function point count. For projects, this might mean that counts for several applications are included.

The application boundary indicates the border between the software being measured and the user. It is determined by the user's view of the application and is critical as it determines what is internal and external to the application.

3.1.3. Determine the Unadjusted Function Point Count

The first step in determining the Unadjusted Function Point Count is to identify the data function types, namely Internal Logical Files (ILFs) and External Interface Files (EIFs). An ILF is a logical group of data maintained by the application (e.g., Employee file in a Human Resource system). An EIF is a logical file that is referenced but not maintained by the application being counted (e.g., Global Province table). Depending on the number of attributes and logical subgroups of data elements in the ILF or EIF, they are ranked as having a Low, Medium or High weight. Each weight for an ILF and EIF is attributed a number of function points (e.g., Low ILF is 7 function points, High EIF is 10 function points).

The second step is to examine the External Inputs (EI), External Outputs (EO) and External Queries (EQ). These are the transactional functions. An EI maintains an ILF or passes control data to the application. An EO presents information to the user with added value such as mathematical calculations or derived data (e.g., report with totals). An EQ also presents information to the user but without any added value. Each EI, EO and EQ is assigned a weight of Low, Medium or High depending on the number of ILFs and EIFs referenced, and the numbers of attributes in the transaction. Each weight for an EI, EO and EQ is associated with a specific number of function points.

Once all the ILFs, EIFs, EIs, EOs and EQs have been identified and weighted, the total of the function points is calculated to determine the Unadjusted Function Point Count.

3.1.4. Determine the Value Adjustment Factor

The Value Adjustment Factor (VAF) adjusts the Unadjusted Function Point Count by +/- 35%. The VAF is intended to take into account some complexities of the application that are not factored into the function point count by evaluating 14 general system characteristics (e.g., data communications, transaction rate, distributed data processing, operational ease, multiple sites, etc.). Each General System Characteristic is ranked from 0 to 5 to determine the VAF.

3.1.5. Calculate the Adjusted Function Point Count

The equation used to determine the Adjusted Function Point Count depends on the type of count (e.g., application vs. development vs. enhancement). The simplest of calculations is for the Application Function Point Count where the Unadjusted Function Point Count is multiplied by the value adjustment factor (e.g., UFPC * VAF). The other equations for Development, Enhancement and Revised Application Counts take into account functionality that was added, changed or deleted and the VAF before and after the changes.

3.2. Types of Function Point Counts and their Accuracy

Function Point Counts can be done at different points throughout the software design and development process. Depending on when the count is performed and the documentation available (e.g., requirements only, use cases, functional design specifications, installed application), different levels of accuracy are obtained. Counts can be updated as more information becomes available but they provide a useful metric throughout the project lifecycle for estimating purposes. The three types of counts that can take place are indicative function point counts, estimated function point counts and detailed function point counts.

This report outlines the indicative function point count for the GCL System.

3.2.1. Indicative Function Point Count

An indicative function point count takes place early on in the project and is based on information provided in either the conceptual data model or the normalized data model, depending on which of the two is available. To perform the indicative count, information is required on whether the entities in the models will be either maintained or referenced by the application being counted (i.e., are they ILFs or EIFs).

If the indicative count is based on the conceptual data model, the number of function points is equal to the total of 35 functions points per ILF and 15 function points per EIF. For an indicative count based on a data model in third normal form, 25 functions points per ILF and 10 functions points per EIF are counted. As described by NESMA, "the multiplication factors are based on the

assumption that a minimum of three external inputs (add, change, and delete), one external output, and one external inquiry will usually be present for each internal logical file, and that a minimum of one external output and one external inquiry will be present for each external interface file".

Note: While performing an indicative function point count can be very helpful in the initial project phase to gauge application size, this type of count should be used with caution as deviations of up to 50% higher or lower are possible.

3.2.2. Estimated Function Point Count

An estimated function point count can take place during or at the end of the requirements phase of the development cycle when the following information is available to the counter:

- A data model that shows the relationships between logical files
- Information on how the logical files are maintained or referenced by the application
- Models that show the incoming and outgoing information flows (e.g., interfaces to other applications)
- Information on the fourteen general system characteristics

By reviewing this information, the data function types (ILFs and EIFs) and the transactional function types (EOs, EQs and EIs) are identified. An average weight for complexity is assigned to each transactional function and a low complexity for each data function. The total is equal to the estimated function point count.

3.2.3. Detailed Function Point Count

The Detailed Function Point Count is the most accurate count and can be completed when all the specifications are known, usually at the end of the analysis and/or design stage. To perform a detailed count, a data model must be available which provides information on all the data elements and user recognizable subgroups of data elements in the ILFs and EIFs. All transactions are known, as well as all the data element types and files referenced during these transactions. Information on the fourteen general systems characteristics is also required to perform a detailed count. Based on this information, the complexity of each of the ILFs, EIFs, EIs, EOs and EQs can be determined. The total is equal to the detailed function point count.

4. INDICATIVE FUNCTION POINT COUNT

This chapter outlines the results of the indicative function point count for the GCL System.

4.1. Indicative Function Point Count Type

This indicative function point count is for a new GCL System. All the GCL System functionality outlined in the GCL Definition of Requirements Appendix C was counted, except for any functionality listed in the assumptions as being out of scope.

4.2. Application Boundary and Scope

The application boundary was deemed to surround the functionality described by the 34 Use Cases found in Appendix C of the GCL Definition of Requirements.

All this functionality was assumed to be part of one system.

4.3. Assumptions and Notes

The following assumptions were made during the indicative function point count. These assumptions are critical as they affect the final function point count:

- Assumed that the conceptual data model entities to be counted are those listed in Appendix A of this report
- Assumed that grants, loans and contracts can be amalgamated into a single Agreement entity type
- Assumed that applicants and respondents can be amalgamated into a single Applicant/Respondent entity type
- Assumed that recipients and contractors can be amalgamated into a single Recipient/Contractor entity type
- Assumed that reference tables cannot be counted per IFPUG Counting Practices Manual 4.1.1. Instead reference tables are counted as one entity type as per NESMA FPA Definition and Counting Guidelines 2.0.
- Assumed that the only external applications that the GCL System interfaces with are the AFRS System and up to 4 Agency Financial Systems, i.e. a total of 5 external applications

- Assumed that no transactional data is imported into the GCL System, and no messages are received by the GCL System from external applications other than acknowledgement of data transmission by the 5 specified external interfacing applications
- Assumed that the only data referenced by the GCL System but maintained in an external application is data related to Financial Coding for Budgets, and data related to Vendors
- Assumed that on-line help functionality is required for the GCL System (see Use Case 27)
- Assumed that error message data must be accessed and maintained for the GCL System
- Assumed that edit criteria data must be maintained to enable configuration of business rules
- Assumed that security data must be maintained for the GCL system (see Use Case 29)
- Assumed that audit data must be captured and maintained for the GCL system

The following should be noted with respect to the indicative function point count. Clarification on these items may significantly affect the final count. (Quotes are taken from the GCL Definition of Requirements report.)

- “The system must meet core and accommodate agency-specific requirements.” The extent of agency-specific requirements could have a significant impact on the size and complexity of the GCL System.
- “Sub-grants and loans especially have a very wide variety of data involved, ranging from wastewater pollutant content to the average income of residents of a building that was built.” The conceptual data model entities identified for the indicative function point count may not be fully representative of the extent of data required to be captured, manipulated and stored in the GCL System.
- The GCL System is heavily dependent on workflow and scheduling components. More information is needed on workflow and scheduling requirements in order to assess the true complexity of these components.
- “The solution application will provide easy to use standard reports.” No requirements for standard reports have been identified to date (other than the Use Case 15: Report to Funding Source). The extent of standard reporting requirements could have a significant impact on the size of the GCL System.
- Data migration/conversion requirements have not been identified at this time, and are therefore not included in scope for the indicative function point count.

4.4. Function Point Summary

The base indicative function point count is 1,185. The level of accuracy for an indicative count is plus/minus 50% giving a FPC range of 593 to 1,778. Given the anticipated high level of business, data and workflow complexity inherent in this type of system, and the estimating risks and

assumptions presented in Section 4.3 of this report, it is recommended that the lower range figure be discounted, and the indicative FPC range formalized at 1,185 to 1,778 function points.

For detailed information on the indicative function point count, please refer to the spreadsheet in Appendix A.

5. CONCLUSION

This report documents the results of an indicative function point analysis of the GCL System. The indicative function point count is based on the GCL Definition of Requirements Appendix C (34 Use Cases). Important assumptions were made during the count and should these be incorrect, the final count may be affected. The count results are provided in Section 4.4 Function Point Summary.

Appendix A. Indicative Function Point Count Spreadsheet

| WA State Office of Financial Management - Grants, Contracts and Loans Feasibility Study | | | | | |
|---|-----------|--------------|------------|--------------|--|
| Indicative Function Point Count - Author: B. Truter - Date: Feb 9, 2006 | | | | | |
| | Count | FPs | min FPs | max FPs | |
| ILF Business Entities | | | | | |
| Agency | | | | | |
| Agency Staff Roles | | | | | |
| Agency Staff | | | | | |
| Agreement | | | | | |
| Advertisement Item | | | | | |
| Application Form | | | | | |
| Applicant/Respondent | | | | | |
| Application | | | | | |
| Application Evaluation Criteria | | | | | |
| Application Evaluation Workflow Group | | | | | |
| Application Evaluation Workflow Events | | | | | |
| Application Evaluation Results | | | | | |
| Application Award | | | | | |
| Recipient/Contractor | | | | | |
| Agreement Schedule | | | | | |
| Agreement Budget | | | | | |
| Agreement Deliverables | | | | | |
| Agreement Outcomes | | | | | |
| Agreement Workflow Group | | | | | |
| Agreement Workflow Events | | | | | |
| Payment Request | | | | | |
| Financial Transactions | | | | | |
| Recipient/Contractor Progress Data | | | | | |
| Inspection/Audit Results | | | | | |
| Close Out Evaluation Results | | | | | |
| Application Formats | | | | | |
| Funder Progress Report Formats | | | | | |
| Reference Tables | | | | | |
| Total ILF Business Entities | 28 | | | | |
| ILF System Entities | | | | | |
| Help Messages | | | | | |
| Error Messages | | | | | |
| Application Security | | | | | |
| Edit Criteria | | | | | |
| Audit Data | | | | | |
| Total ILF System Entities | 5 | | | | |
| Total ILFs (ILF multiplier = 35) | 33 | 1,155 | | | |
| EIF Business Entities | | | | | |
| Financial Coding for Budgets | | | | | |
| Vendor Data | | | | | |
| Total EIFs (EIF multiplier = 15) | 2 | 30 | | | |
| Total Indicative Function Point Count (range +/- 50%) | | 1,185 | 593 | 1,778 | |